

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2003-114817

(43)Date of publication of application : 18.04.2003

(51)Int.Cl.

G06F 12/00
G11B 7/007
G11B 20/12
H04N 5/85

(21)Application number : 2001-342145

(71)Applicant : TOSHIBA CORP

(22)Date of filing : 07.11.2001

(72)Inventor : ANDO HIDEO
TAKAHASHI HIDEKI
MIMURA HIDENORI

(30)Priority

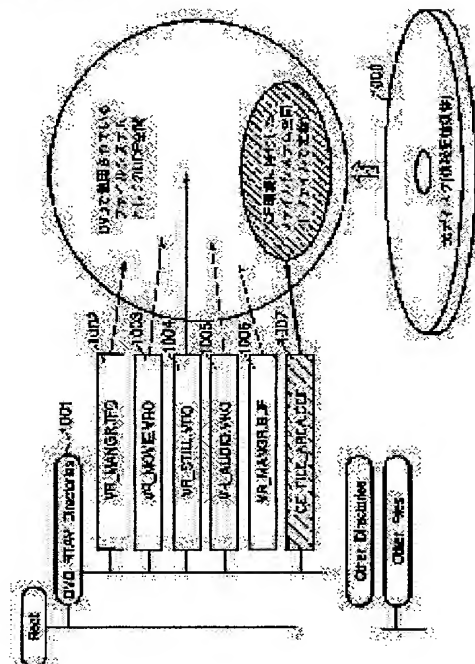
Priority number : 2001236731 Priority date : 03.08.2001 Priority country : JP

(54) INFORMATION STORAGE MEDIUM AND INFORMATION RECORDING METHOD

(57)Abstract:

PROBLEM TO BE SOLVED: To provide an information storage medium which can make use of individual advantages of a plurality of file systems.

SOLUTION: An information storage medium (1) has a logical space managed by a predetermined file system, wherein the logical space is managed by either one of two different types of file systems and a file (1007) to designate a space area of the other file system is provided in the logical space.



特開2003-114817
(P2003-114817A)

(43)公開日 平成15年4月18日(2003.4.18)

(51)IntCl. ⁷	識別記号	FI	チーフ・イニシアチブ(参考)
G 0 6 F 12/00	5 2 0	G 0 6 F 12/00	5 2 0 J 5 B 0 8 2
G 1 1 B 7/007	20/12	G 1 1 B 7/007	5 C 0 5 2
			5 D 0 4 4
H 0 4 N 5/85		H 0 4 N 5/85	Z 5 D 0 9 0

審査請求 有 請求項の数 2 OL (全 23 頁)

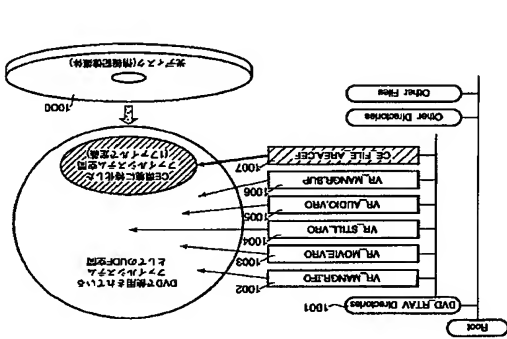
(21)出願番号	特願2001-342145(P2001-342145)	(71)出願人	00003078 株式会社東芝
(22)出願日	平成13年11月7日(2001.11.7)	(72)発明者	安東 秀夫 神奈川県横浜市西区新町70番地 株式会社 東芝柳町事業所内 高橋 秀樹 神奈川県横浜市西区新町70番地 株式会社 東芝柳町事業所内
(31)優先権主張番号	特願2001-236731(P2001-236731)	(74)代理人	100058479 弁理士 鈴江 武彦 (外6名)
(32)優先日	平成13年8月3日(2001.8.3)		
(33)優先権主張国	日本 (JP)		

(54) 【発明の名称】 情報記憶媒体及び情報記録方法

(57) 【要約】

【課題】複数のファイルシステムの夫々の長所を生かすことが可能な情報記憶媒体を提供すること。

【解決手段】所定のファイルシステムで管理される論理空間を有する情報記憶媒体(1)であって、異なる2種類のファイルシステムのどちらから一方のファイルシステムで前記論理空間が管理され、この論理空間上に他方のファイルシステムの空間領域を指定するためのファイル(10007)を有する。



(2) 特開2003-114817

格により定義されているファイルだけでなく、ワープロや表計算ソフトのファイルなどを一つのディレクトリで統一的に扱う態様を提供することが望まれている。AV専用のファイルシステムではPCファイルの取り扱いが煩雑になる。

【0008】この発明の目的は、上記問題を解決するために、複数のファイルシステムの夫々の長所を生かすことが可能な情報記憶媒体及び情報記録方法を提案することにある。

【0009】つまり、同一の情報記憶媒体上でCE市場に特化した簡単な構造を持つファイルシステムの採用を可能とし、同時に同じ情報記憶媒体上で既存のDVDアプリケーションファイル及びPCファイルの取り扱いが可能とする。

【0010】

【課題を解決するための手段】上記課題を解決し、目的を達成するために、この発明の情報記憶媒体及び情報記録方法は、以下のように構成されている。

【0011】(1) この発明の情報記憶媒体は、所定のファイルシステムで管理される論理空間を有し、異なる2種類のファイルシステムのどちらから一方のファイルシステムで前記論理空間が管理され、この論理空間上に他方のファイルシステムの空間領域を指定するためのファイル(10007)を有する。

【0012】(2) この発明の情報記録方法は、所定のファイルシステムで管理される論理空間を有する情報記憶媒体に対して、異なる2種類のファイルシステムのどちらから一方のファイルシステムで前記論理空間を管理し、この論理空間上に他方のファイルシステムの空間領域を指定するためのファイル(10007)を有する。

【0013】

【発明の実施の形態】まず、この発明のポイントを列記する。

【0014】(1) UDFとCE市場に特化した独自のファイルシステムと異なる2種類のファイルシステムを同一情報記憶媒体上に共存記録させる。

【0015】(2) 異なる2種類のファイルシステムのどちらから一方のファイルシステムで情報記憶媒体上の論理空間を管理させ、その論理空間上に他方のファイルシステム空間領域を指定するためのファイル(10007)を有する。

【0016】(3) AV特化のファイルアプリケーションに関して強い制限を設けることでCEの負荷を軽減する。

【0017】(4) UDFのファイルシステムドライバを持たなくてもUDF上のファイル操作が可能な仕組みを得る。

【0018】以下、この発明の実施の形態について図面を参照して説明する。

【0019】図1は、CE環境に特化したファイルシステムとDVDで使用されているUDFとの間の共存方法

を示している。光ディスク（情報記憶媒体）1000上のDVD用のファイルはUDFに特別に割り当てられたディレクトリに格納される。ここでは記録系アプリケーションファイルが格納するために定義されたDVD_RTA V Directories 1001に複数のファイルが置かれていた状態を示している。

[0020] この実施例では従来のVideo記録用ファイル群（VR_MANGR、IFO 1002、VR_Movie、VRO 1003、VR_STILL、VRO 1004、VR_AUDIO、VRO 1005、VR_MANGR、BU P 1006）に加え新規に定義されるファイルCE_FILE_AREA、CEF 1007が置かれている。従来のVideo記録用ファイル群はUDFで直接管理される。それに対して、CE_FILE_AREA、CEFはファイルの占有するエリアはUDF上に定義されるもののUDFとは別の領域に特化したファイルシステム空間を内部に持っていることを特徴とする。このCE_FILE_AREA、CEFはファイルではあるが1つのパーティションのような機能を持つ。すなわち、このファイルはコンテンツサイズに応じ変わるのではなく、まずは領域として確保されるだけである。この領域確保の時にUDF上にファイル登録される必要があり、いったんUDF上にファイル登録された後は、このファイル内の情報を書き換えてもファイルサイズ及び占有しているアドレス位置は変化しない。すなわち、UDF上は不変の領域となる。もし、ファイルサイズを変更する場合には以降に示すように特別のプロセスを経ることが必要となる。

[0021] 図2は、図1で示した方法において、UDF論理アドレス空間上で各種ファイルの配置設定方法を比較したものである。まず、PCファイルとDVDアプリケーションの管理ファイルについて、ファイルアロケーション上はセクタ単位の割り当てで、この例のセクタサイズは2KBである。次にDVDオブジェクトファイル（画像用ファイル）は不連続部分を含有する連続領域であるCDA（Contiguous Data Area）の最小サイズが2MBであり、CDA中に欠陥セクタ等が含まれる場合には連続性がとぎれてしまう。このため、再生時のバッファアランダーランを防止するためにCDAサイズを調整する。例えば、欠陥セクタ、ゾーンバウンダリ、他のファイルが入る、場合がある。CDAサイズを計算するためにはこれらの不連続点の数（セクタ数）と分布を考慮することが必要である。記録時に新たに発見された欠陥が無い場合には、記録位置に関する情報をホストにあらかじめ入手できることでCDAサイズを計算することが可能である。記録時に新たに発見された欠陥をスキップする場合には発見された時点で必要十分なCDAサイズを求めるためには再計算が必要となる。このようにリアルタイム処理の方法についてはセットの作り方

によるが、記録された結果が、再生時にバッファアランダーランが発生させない状態になっていなければならない。

[0022] DVDオブジェクトファイルの場合には記録時にはシーマレス再生を保证するようなアロケーションが必要となっているが、記録後に編集した場合にはシーマレスは条件とならない。例えば、記録データの一部を削除する場合には2KB単位（セクタ）で処理することが可能である。また、CDAとUDF上のExtent（ファイルの部分）であり、1つ以上のExtentを結合して1ファイルが構成される）との関係は、1つのCDAは1つ以上のExtentで構成される。これはExtentが定義上連続であることになっているから、不連続箇所を含むとその場所でのExtentの分割が必要となるからである。また、DVDオブジェクトファイルの場合にはCDAの境界位置はいくらでも動きうる。

[0023] 図3は、図1で示した方法でのUDF論理アドレス空間上のDVDオブジェクトファイルの配置設定方法の例である。DVD Object File 位置を指定するCDA（あるいはExtent）のサイズ（K、M、N）は、 $K \geq 2MB$ 、 $M \geq 2MB$ 、 $N \geq 2MB$ を満足する限り、2KBの整数倍として任意の値に設定される。CDAサイズとExtentサイズは必ずしも一致する必要は無く、1個以上のExtentから1個のCDAを構成する。2KB単位でのファイル内の追加/解放が可能である。例えば、CDA #3に含まれるデータ（サクション1）の部分削除に伴い、CDA #3が、CDA #3aとCDA #3bに分割されることがある。なお、削除データのサイズN1+CDA #3aのサイズN2+CDA #3bのサイズN3=N、且つN2 \geq 2MB、N3 \geq 2MBの関係が成り立つ。

[0024] これに対してCE環境に特化したファイルシステム空間を有するファイル内では、DVDオブジェクトファイルの基本単位であるCDAを固定サイズ（4MB以上）として扱う。ただし、CDAサイズをあまり小さくすると再生時のバッファアランダーランが発生するので、例えば4MBとする。更に大きなサイズを割り当てても可能であるが、基本単位があまり大きいと今度ではディスクの使用効率が悪くなる。このCE特化のファイルシステム内ではPCファイルが入り込むことは無いのであらかじめ固定サイズ単位の領域をあらかじめ設定しておくことが可能である。例えば4MB単位とした場合には、CDAの境界は4MB単位に存在し変化したくない。言い方を要えと、このファイルサイズは4MBの倍数になっている。

[0025] 図4は、図1に示した方法でのUDF論理アドレス空間上でのCE環境に特化したファイルシステム空間を表すファイルの配置設定方法の例である。CE_FILE_AREA、CEFファイルのFile Entry = FE（AD（L、D）、AD（L、E）、AD（L、

E+L）、AD（L、F）、AD（L、F+2L））CDAサイズとExtentサイズは等しく、全てのCDAサイズが一定（4MB以上）。CE環境に特化したファイルシステム空間内での疑似ファイル（AV_FILE_01、MPG/AV_FILE_02、MPG）サイズは、CDAのサイズの整数倍とし、ディスク上の配置場所はCDAの配置場所と一致させる。AV疑似ファイルは固定長CDA単位での追加/解放を行う。

[0026] 図5は、図1に示した方法での各種ファイルと疑似ファイルの表示説明図である。ここに示したようにCE_FILE_AREA、CEFについては、内部はファイル構成をとっており、PC上でこのファイル内のCEファイル群を見るためには、特別のファイルViewerを介する必要がある。また、このようにCE_FILE_AREA、CEF内はファイル群で構成されているので、必要があれば、各ファイルをUDFから直接見える領域に取り出すことが可能である。一度UDFから直接見える場所に取り出されたファイルなら通常のPC上のファイル操作でコピー、移動操作することが可能となる。また、逆にCE_FILE_AREA、CEFに戻すことも可能である。基本的に、シーマレス再生を行うためには、CE_FILE_AREA、CEFが必要であるため、CEファイル上のアロケーションに工夫に取り込む時点で、特別なアロケーションツール介しを行うことになる。

[0027] ここで、CE環境に特化したファイルシステムとUDFの両者のメリットの比較、及び同一記憶媒体上に両者を共存させる意義についてまとめる。

[0028] CE環境に特化したファイルシステムを採用するメリットとしては以下のようものが挙げられる。

[0029] (1) 使用目的をAV情報記録に特化させ、最適で簡素なファイルシステムにして、制御ソフトウェアのコンパクト化を図ることができる。このファイルシステム内でのPCファイルの混在記録を禁止し、制御ソフトウェアを簡素化する。

[0030] (2) CE環境に特化し、コンパクトなファイルシステムコンポーネントを構築し、上位レイヤーでの統合活用を可能にする。上位レイヤーでのPCファイルとの共存を可能とする。

[0031] ファイルシステムとしてUDFを採用するメリットとしては以下のようものが挙げられる。

[0032] (1) 現行DVD規格に基づいて作成するAV情報の記録が可能となる。

[0033] (2) PCファイルの混在記録が可能となる。

[0034] (3) 既に製品化されたDVD関連製品のファイルシステムに関する制御部の有効活用が可能となる。

[0035] 同一記憶媒体上に両者を共存させることにより、両者のメリットを引き出すことができる。

[0036] CE_FILE_AREA、CEFをアロケートするときの制限として次のようなものが考えられる。

[0037] (1) 再配置を禁止する

例えば、ファイルシステムレベルでデフラグなどを行うと配置がずれてしまうので、ファイルタイプ249あるいはDVD専用に新規番号の割り当て、Non-Relocableに設定しておくなど、UDF上の特別な管理が必要である。UDFに対して再配置が禁止されれば、割り当てられた領域が変化しないのでLSN（Logical Address Number）などファイルとして割り当てられている領域情報をファイル内に持つて、その中でだけ処理することが可能となる。

[0038] (2) ファイルのスタートアドレスを固定にする

ファイルのスタートアドレスを固定にすることができればUDFを解釈することなく、CEは記録再生処理を行うことができる。

[0039] (3) CE_FILE_AREA、CEFの構成を示すファイルを別定義して配置

全Extentの位置情報を1ファイルに集めて置く。

[0040] 図6は、CE_FILE_AREA、CEFのファイルサイズの拡大（縮小）例である。ファイルは、1つ以上の領域によって構成されており、それぞれは、固定長1の倍数を単位としてサイズが可変になっている。

[0041] 図7は、UDFによる従来のVolume&File Structure（ボリューム&ファイルストラクチャー）の読み出し手順である。これに対し、図8で示したようにCE_FILE_AREA、CEFを適用に設定することでUDFの解釈が大幅に減らして、Volume&File Structureを認み出すことが可能となる。ここでは、固定LSNに配置したTrigger（トリガー）ファイル定義することで、このTriggerファイルをサーチしてUDFの解釈を不要にした例を示している。Triggerファイルとは、図10に示したようにCE_FILE_AREA、CEFの位置情報を示したものである。このTriggerファイルは定義することにより範囲はUDFを解釈しなくてもファイルの位置が特定できる。Triggerファイルの解釈は必要であるが、UDFを解釈することによって遙かに容易である。

[0042] 図11～図13にTriggerファイルの活用例を示した。例えばTriggerファイルであるか否かを確認するために定義している識別子256バイトがあれば識別誤りが発生する確率が十分小さい。このバイト数は必要に応じて小さくしてもよい。Triggerファイルはあらかじめ決めた固定LSNに配置されておきそこからMセクタ分をTriggerファイル

領域と決めておき、Triggerファイルの内容を複数回、異なるECCブロックに配置しておけば何らかの障害が発生してファイルの先頭が読み出せなくなっても情報取り出しが可能である。

【0043】図11に示すように、Triggerファイルの位置が確認される。まず、固定LSNからはじめのセクタが読み込まれる(ST11)、識別子(RBP0～255)が調べられる(ST12)。Triggerファイルであることが確認されると(ST13)、分割領域数(RBP256～257)が確認され(ST14)。必要4、ファイルの位置が確認される(ST15)。必要に応じてセクタ目以降が読み込まれる(ST16)。【0044】続いて、図12を参照して、はじめのセクタが読み出せない場合には、次のECC(Error Correction Code)ブロックの先頭セクタが読み出される(ST21)。規定のアドレスを超える場合には(ST22、YES)、エラーとなる(ST23)。規定のアドレス以下であれば(ST22、NO)、識別子(RBP0～255)が調べられる(ST24)。Triggerファイルであることが確認されると(ST25、YES)、分割領域数(RBP256～257)が確認され(ST26)、ファイルの位置が確認される(ST27)。必要に応じてセクタ目以降が読み込まれる(ST28)。

【0045】続いて、図13を参照して、領域の追加削除を行う場合の処理について説明する。追加削除可能な単位に注意してUDF上で領域追加削除が処理され(ST31)、これにあわせてTriggerファイルが変更される(ST32)。

【0046】図9では、CE_FILE_AREA、C_EFの割り当てとWRITE/READ処理方法について示している。CE_FILE_AREA、CEFははじめにUDFでLBN(Logical Block Number)内に割り当てられる。割り当てられた後、ファイルの位置情報をTriggerファイルに記録する。この例ではTriggerファイルもUDFに配置されたファイルであり、Triggerファイルの置かれていた位置はあらかじめ決められた固定LSNから始まるあらかじめ決められたMセクタ分である。このようにいったんTriggerファイルが生成されれば、CE装置ではUDFを解除するのではなくTriggerファイルからCE_FILE_AREA、CEFを認識して(ST91)、アプリケーション内で使われるアドレス(ADAP: Address in Application)を決定し(ST92)、このADAPをLSNに変換し(ST93)、変換されたLSNによりWRITE/READが可能となる(ST94)。

【0047】自由度を大きくするためにTriggerファイルは実体のファイルであるCE_FILE_AR

EA、CEFから分離した。しかし、CE_FILE_AREA、CEF内(例えば先頭)にTrigger_FILE_AREA、CEFの先頭LSNが固定であればCE装置にとって有利である。

【0048】上記の仕組みをシステムとして構成した例が図14である。ホストではCE_FILE_AREA、CEFを扱うためのLocal FSドライバを特許ライブラリコマンド制御部を介してドライバをコントロール可能である。例えば、はじめにディスクの全領域をCE_FILE_AREA、CEFとしてしまえばそれ以降全くUDF管理を行うことなく簡易なシステム制御が可能となる。

【0049】また、CE_FILE_AREA、CEFをDVD_RTA V Directories1001全体に予めアサインすることにより、UDFの変更を必要にすることもできる。

【0050】以下、図17～図19に示すフローチャートを参照して、CE_FILE_AREA、CEF空間に対する記録、CE_FILE_AREA、CEF空間の追加、CE_FILE_AREA、CEF空間の縮小などについて説明する。

【0051】最初に、図17に示すフローチャートを参照して、CE_FILE_AREA、CEF空間への記録処理についてまとめると、まず、Triggerファイルを参照の上、未記録CDAが探され(ST41)、未記録CDAに対してスタートポイントがセットされ(ST42)、記録が開始される(ST43)。未記録CDAの残容量が所定容量を下回ると(ST44、NO)、追加処理(Triggerファイル再設定)が実行される(ST46)。記録が継続されている間(ST45、YES)、ST44～ST46の処理が繰り返される。【0052】続いて、図18に示すフローチャートを参照して、未記録CDAの追加処理をまとめると、まず、アプリケーションエンジンからUDFドライバに対して追加が指示される(ST51)。これに伴いファイルの実体が拡張される(ST52)。つまり、Triggerファイルが変更される。

【0053】続いて、図19に示すフローチャートを参照して、CE_FILE_AREA、CEFのサイズ縮小処理について説明する。CE_FILE_AREA、CEFのサイズの縮小は、例えば、UDF上のデータが満杯になったときに実行される。まず、アプリケーションエンジンからUDFドライバに縮小が指示される(ST61)。これに伴いCE_FILE_AREA、CEFが解放され空きが調査される(ST62)、ファイルの実体が縮小される(ST63)。つまり、Triggerファイルが変更される。

【0054】図15に図1とは異なる本発明の他の実施例を説明する。図1ではUDF空間をベースとし、その

中にファイル定義によりC区領域に特化したファイルシステム空間を構築した。それに対して図15では光ディスク(情報記憶媒体)上の全論理空間を“CE領域に特化したファイルシステム空間”に割り当て、CE領域に特化したファイルシステム空間をベースにする。そして“CE領域に特化したファイルシステム空間”内に例えば“UDF_FILE_AREA、UDF”と言う名のファイルを定義して、そのファイルの配置場所としてD VDで使用されているファイルシステムとしての“UDF空間”を割り当てる。

【0055】図15に示した方法での各ファイルの記録方法を図16に示した。図15の方法を採用した場合に図16に示すように、光ディスク(情報記憶媒体)上の全論理アドレス空間上に固定サイズ(例えば4MB以上の)CDAで等分割し、記録ファイルの配置場所をそのCDA位置を基準として割り当てる所に特徴がある。すなわちUDF空間を割り当てるためのファイルである“UDF_FILE_AREA、UDF 2006”の位置とサイズはこの固定サイズで有る1個のCDAまたは複数のCDAを割り付ける。

【0056】図16に示した実施例ではCDA#24～CDA#26の場所を割り当てている。すなわちこのCDA#24～CDA#26の範囲内がUDF空間として確立される。CDA#24～CDA#26の範囲内を更に2k B単位で分割してUDF空間内のアドレスである相対的なLBN(Logical Block Number)を割り当てる。DVDのObject File(例えば“VR_MOVIE.VRO”ファイル)はこのUDF空間内には配置可能となる。図2に示したようにDVD Objectfileに対するCDAは2MB以上の可変(但し2k B単位での変換)サイズで割り当てられ、このファイルのFile Entry内のAllocation Descriptorで示されるExtentは上記の相対的なLBN(Logical Block Number)を用いて記述される。

【0057】CE領域に特化したファイルシステム空間上での一般ファイル位置も固定サイズのCDAを基準として設定される。図15における“AV_FILE_01.MPG 2004”と“AV_FILE_02.MPG 2005”がAVファイル(映像または音声情報)に関係したObject File)を意味し、“TOC_FILE_1”と“AV_MANG.MNG 2002”がそのObject Filesに関する管理情報が記録された管理ファイルになっている。Object Fileで有る“AV_FILE_01.MPG 2004”及び“AV_FILE_02.MPG 2005”に対する属性情報が記録されたファイルが“AV_MANG.MNG 2002”で有った。全体のTOC(Table of Contents:各オブジェクト間の再生手順を示す)に関する情報が“TOC_FILE_1.IFO 2001”内に記録されている。【0058】ここで、上記したこの発明の構成及び作用

効果についてまとめると、

【0059】(1)この発明の情報記憶媒体は、所定のファイルシステムで管理される論理空間を有し、異なる2種類のファイルシステムとのどちらか一方のファイルシステムで前記論理空間が管理され、この論理空間上に他のファイルシステム空間領域を指定するためのファイルを有する。

【0060】(2)この発明の情報記憶方法は、所定のファイルシステムで管理される論理空間を有する情報記憶媒体に対して、異なる2種類のファイルシステムのどちらか一方のファイルシステムで前記論理空間を管理し、この論理空間上に他のファイルシステムの空間領域を指定するためのファイルを記録する。

【0061】上記構成により、下記の作用効果が得られる。

【0062】(1)追記(境界位置の移動含む)処理事前に2種類のファイルシステム領域を設定するがファイルの追加を繰り返すと一方の領域が狭くなり、2種類のファイルシステム領域間の境界位置を移動させる必要が生ずる。本発明では情報記憶媒体上の論理空間を管理するファイルシステム側で他のファイルシステム空間領域を指定するためのファイルサイズを変更するだけで容易に境界位置の移動が可能となる。つまり、追記(境界位置の移動含む)処理の容易性が高い。

【0063】(2)既存DVDアプリケーション規格上のAVファイルとCE市場に特化した独自のファイルシステム上に構築されるアプリケーションファイル間の記録再生

2種類のファイルシステム領域が完全に分離され、基本的に一方のファイルシステム上の記録・再生処理が他方のファイルシステムに影響を及ぼす事が内での両者間の記録再生の容易性が高い。

【0064】(3)PCアプリケーションファイル既存のDVD規格に準拠した情報記憶媒体(光ディスク)で実装されているようにUDFで定義された論理空間上にPCアプリケーションファイルを記録することは容易となる。

【0065】(4)既存のライティングソフト既存のDVDプレーヤーやDVDレコーダーはUDF上で処理できるようにシステムアップされているので、UDF領域に関して既存のDVDライティングソフトで記録可能となる。

【0066】(5)既存のファイルシステムドライバ既存のDVDプレーヤーやDVDレコーダーはUDF上で処理できるようにシステムアップされているので、UDF領域に関して既存の既存のファイルシステムドライバを使用する。

【0067】(6)アプリケーションジョック本発明では情報記憶媒体上の論理空間を管理するファイルシステム側で“ファイル位置指定”の形で他のファ

11
 イルシステム空間領域を指定するためアロケーションロ
 ジックが単純化される。
 【0068】(7) ファイルシステムドライバへの簡
 素化
 ファイルシステムドライバの簡素化を図ることがで
 きる。
 【0069】(8) テスト
 現存のDVDプレーヤーやDVDレコーダーはUDF上
 で処理できるようにシステムアップされているので、U
 DF空間上でのテストソフトとして既存のソフトが使用
 できる。UDF空間上でのテストが非常に容易とな
 る。
 【0070】(9) 部分消去とFS管理
 アプリケーション上の部分消去とFS管理への連携容易
 性が確保されている。

10
 【0071】次に、図20～図29を用いてファイルシ
 ステムの方式としてのUDFについて説明する。

【0072】
 <<<UDFの概要説明 (UDFとは何か)>>>
 <<UDFとは何か>>>UDFとは Universal Disk Fo
 rmat の略で、主にディスク状態情報記憶媒体における
 “ファイル管理方法に関する規約”を示す。CD-ROM、
 M、CD-R、CD-RW、DVD-Video、DVD-R
 OM、DVD-R、DVD-RAMは“ISO9660”で規格化されたUDFフォーマットを採用してい
 る。

20
 【0073】ファイル管理方法としては基本的にルー
 トディレクトリ (Root Directory) を割に持ち、ツリ
 ー状にファイル管理する階層ファイルシステムを前提
 としている。
 【0074】ここでは主にDVD-RAM規格 (File
 System Specifications) に準拠したUDFフォーマッ
 トについての説明を行うが、この説明内容の多くの部分
 はDVD-ROM規格内容とも一致している。
 【0075】<<UDFの概要>>
 <階層記憶媒体へのファイル情報記録内容>階層記憶媒
 体には情報を記録する場合、情報のまとまりを“ファイ
 ル” (File Data) と呼び、ファイルデータ単位で
 記録を行う。他のファイルデータと識別するためファイ
 ルデータ毎に独自のファイル名が付加されている。共通
 な情報内容を持つ複数ファイルデータ毎にグループ化す
 るとファイルデータ毎のグループを“ディレクトリ” (D
 irectory) または“フォルダー” (Folder) と呼
 ぶ。各ディレクトリ (フォルダー) 毎に独自のディレク
 トリ名 (フォルダー名) が付加される。更にその複数の
 ディレクトリ (フォルダー) を集めて、その上の階層の
 グループとして上位のディレクトリ (上位フォルダー)
 でまとめる事が出来る。ここではファイルデータとディ
 レクトリ (フォルダー) を総称してファイル (File)

13
 1 個の ルートディレクトリ (Root Directory) 401
 が存在し、その下にサブディレクトリ (SubDirector
 y) 402 が属している。この SubDirectory 402 の中に
 ファイルデータ (File Data) 403 が存在している。
 【0083】実際にはこの例に限らず Root Directory
 401 の直接下に File Data 403 が存在したり、複数の
 SubDirectory 402 が直列につながった複雑な階層構造
 を持つ場合もある。
 【0084】<情報記憶媒体上ファイル管理情報の記録
 内容>ファイル管理情報は上述した論理ブロック単位で
 記録される。各論理ブロック内に記録される内容は以下
 の通りである。
 【0085】・ファイルに関する情報を示す記述文 F
 ID (File Identifier Descriptor)
 ファイルの履歴やファイル名 (Root Directory 名、Su
 bDirectory 名、FileData 名など) を記述している。
 【0086】F IDの中にそれに続く File Data のデ
 ータ内容や、Directory の中味の記録場所を示す記述文
 (つまり該当ファイルに対応した以下に説明する F E
 の記録位置も記述されている)。
 【0087】・ファイル中味の記録位置を示す記述文 F
 E (File Entry)
 File Data のデータ内容や、Directory (SubDirectory
 などの) 中味に関する情報が記録されている情報記憶
 媒体上の位置 (論理ブロック番号) などを記述してい
 る。

20
 【0088】File Identifier Descriptor の記述内容
 の格納を図25に示した。またその詳細の説明は<<Fi
 le Identifier Descriptor>>で行う。File Entry の
 記述内容の格納は図24に示し、その詳細な説明は<<
 File Entry>>で行う。

30
 【0089】情報記憶媒体上の記録位置を示す記述文は
 図21に示す ロングアロケーションディスタクリプター
 (Long Allocation Descriptor) と図22に示す ショ
 ートアロケーションディスタクリプター (Short Allocati
 on Descriptor) を使っている。それぞれの詳細説明は
 <Long Allocation Descriptor>と<Short Allocation
 Descriptor>で行う。
 【0090】例として図20 (a) のファイルシステ
 ム構造の情報を情報記憶媒体に記録した時の記録内容を図
 20 (b) に示す。図20 (b) の記録内容は以下の通
 りとなる。
 【0091】・論理ブロック番号 “1” の論理ブロック
 に Root Directory 401 の中味が示されている。
 【0092】図20 (a) の例では Root Directory 40
 1 の中には Sub Directory402 のみが入っている。で
 はないが、Root Directory402 の中味として Sub Directory 402
 に関する情報が File Identifier Descriptor 文 404で
 記載されている。また図示して無いが同一論理ブロック内
 に Root Directory 401 自身の情報も File Identifier

Descriptor 文で並記して有る。
 【0093】この Sub Directory 402 の File Identif
 ier Descriptor 文 404 中に Sub Directory 402 の中
 味が何処に記録されているかを示す File Entry 文 405
 の記録位置 (図20 (b) の例では4番目の論理プロ
 ック) が Long Allocation Descriptor 文で記載 (LAD
 2) している。

30
 【0094】・論理ブロック番号 “2” の論理ブロック
 に Sub Directory 402 の中味が記録されている位置を
 示す File Entry 文 405 が記録されている。
 【0095】図20 (a) の例では Sub Directory 402
 の中には File Data 403 のみが入っている。で、Sub
 Directory 402 の中味として実質的には File Data 403
 に関する情報が記録されている File Identifier Desc
 riptor 文 406 の記録位置を示す事になる。

40
 【0096】File Entry 文中の Short Allocation Des
 criptor 文で3番目の論理ブロックに SubDirectory 40
 2 の中味が記録されている事 (AD(3)) が記述されて
 いる。

50
 【0097】・論理ブロック番号 “3” の論理ブロック
 に Sub Directory 402 の中味が記録されている。
 【0098】図20 (a) の例では Sub Directory 402
 の中には File Data 403 のみが入っている。で、Sub
 Directory 402 の中味として File Data 403 に関する
 情報が File Identifier Descriptor 文 406 で記載
 されている。また図示して無いが同一論理ブロック内に
 Sub Directory402 自身の情報も File Identifier Des
 criptor 文で並記して有る。

60
 【0099】File Data 403 に関する File Identifier
 Descriptor 文 406 の中にそのFile Data 403 の内
 容が何処に記録されている位置を示す File Entry 文 4
 07の記録位置 (図20 (b) の例では4番目の論理プロ
 ックに記録されている) がLong Allocation Descriptor
 文で記載 (LAD(4)) されている。

70
 【0100】・論理ブロック番号 “4” の論理ブロック
 に File Data 403 内容408、409が記録されている位置
 を示す File Entry 文 407 が記録されている。

80
 【0101】File Entry 文 407 内の Short Allocati
 on Descriptor 文で File Data 403内容408、409が5番
 目と6番目の論理ブロックに記録している事が記述 (A
 D(5),AD(6)) されている。

90
 【0102】・論理ブロック番号 “5” の論理ブロック
 に File Data 403 内容情報 (a)408が記録されている。
 【0103】・論理ブロック番号 “6” の論理ブロック
 に File Data 403 内容情報 (b)409が記録されている。
 【0104】<図20 (b) 情報に沿った File Data
 へのアクセス方法>
 “<情報記憶媒体上のファイルシステム情報記録内容
 >”で簡単に説明したように File Identifier Descriptor
 文 404、406 と File Entry 405、407 には、それに統

(9) 特開2003-114817 16

(File Allocation Table) の記録場所はあらかじめ決まっているので記録媒体の高い信頼性 (欠陥領域が少ない事) が前提となる。

[0113] UDFではファイル管理情報が分散配置されているので、ファイル構造の大幅な変更が少なく、階層の下部分 (主に Root Directory より下の部分) で後から新たなファイル構造を付け足して行く用途 (主に追加用途) に適している。(追記時には以前のファイル管理情報に対する変更箇所が少ないため。) また分散されたファイル管理情報の記録位置を任意に指定できるの、先天的な欠陥箇所を避けて記録する事が出来る。

[0114] ファイル管理情報を任意の位置に記録できるので全ファイル管理情報を一箇所に集めて記録し上記FATの利点も出せるので、より汎用性の高いファイルシステムと考えることが出来る。

[0115] <<<UDFの各記述文 (Descriptor) の具体的な内容説明>>>

<<<論理ブロック番号の記述文>>>

<Allocation Descriptor>

20 “<情報記録媒体上のファイルシステム情報記録内容>” に示したように File Identifier Descriptor や File Entry などの一部に含まれ、その後に続く情報が記録されている位置 (論理ブロック番号) を示した記述文をAllocation Descriptorと呼ぶ。Allocation Descriptor には以下に示すLong Allocation Descriptor と Short Allocation Descriptor がある。

[0116] <Long Allocation Descriptor>

図21に示すように

- Extent の長さ 410 ... 論理ブロック数を 4Bytes で表示
- Extent の位置 411 ... 該当する論理ブロック番号を 4Bytes で表示
- Implementation Use 412 ... 演算処理に利用する情報で 8Bytes で表示などから構成される。

[0117] この説明文では記述を簡素化して “LAD (論理ブロック番号)” で記述する。

[0118] <Short Allocation Descriptor>

図22に示すように

- Extent の長さ 410 ... 論理ブロック数を 4Bytes で表示
- Extent の位置 411 ... 該当する論理ブロック番号を 4Bytes で表示

のみで構成される。

[0119] この説明文では記述を簡素化して “AD (論理ブロック番号)” で記述する。

[0120] <<Unallocated Space Entry>>図23に示すように情報記録媒体上の “未記録状態の Extent 分布” をExtent毎に Short Allocation Descriptor で記述し、それを並べる記述文で、SpaceTable (図27～図29参照) に用いられる。具体的な内容としては以

く情報が記述して有る論理ブロック番号が記述して有る。Root Directory から階層を下りながら SubDirectory を経由して File Data へ到達すると同様、 File Identifier Descriptor と File Entry 内に記述して有る論理ブロック番号に従って情報記録媒体上の論理ブロック内の情報を順次再生しながらFile Data のデータ内容へアクセスする。

[0105] つまり図20 (b) に示した情報に対して File Data 403へアクセスするには、まず始めに1番目の論理ブロック情報を読む。File Data 403は Sub Directory 402の中に存在しているの、1番目の論理ブロック情報の中から Sub Directory 402の File Identifier Descriptor 404を探し、LAD(2)を読み取った後、それに従って2番目の論理ブロック情報を読む。2番目の論理ブロックには1個の File Entry 文しか記述していないので、その中の AD(3)を読み取り、3番目の論理ブロックへ移動する。3番目の論理ブロックでは File Data 403に関して記述して有る File Identifier Descriptor 406を探し、LAD(4)を読み取る。LAD(4)に従い4番目の論理ブロックへ移動すると、そこには1個の File Entry 文 407しか記述してないので、AD(5)とAD(6)を読み取り、File Data 403の内容が記録して有る論理ブロック番号 (5番目と6番目) を見つける。

[0106] AD (*), LAD (*) の内容については “<UDFの各記述文 (Descriptor) の具体的な内容説明>” で詳細に説明する。

[0107] <<UDFの特徴>>

<UDF特徴説明>以下にHDDやFDD、MOなどで使われているFATとの比較によりUDFの特徴を説明する。

[0108] 1) (最小論理ブロックサイズ、最小論理セクタサイズなどの) 最小単位が大きく、記録すべき情報量の多い映像情報や音楽情報の記録に向く。

[0109] FATの論理セクタサイズが512 Bytes に対して、UDFの論理セクタ (ブロック) サイズは2048 Bytes と大きくなくっている。

[0110] 2) FATはファイルの情報記録媒体への割り当て管理表 (File Allocation Table) が情報記録媒体上で局所的に集中記録されるのに対し、UDFではファイル管理情報をディスク上の任意の位置に分散記録できる。

[0111] UDFではファイル管理情報やファイルデータに関するディスク上の記録位置は論理セクタ (ブロック) 番号として Allocation Descriptor に記述される。

[0112] FATではファイル管理領域 (File Allocation Table) で集中管理されているため頻繁にファイル構造の変更が必要な用途 (主に頻繁な書き換え用途) に適している。(集中箇所に記録されているので管理情報を書き換え易いため。) またファイル管理情報

(10) 特開2003-114817 18

[0133] 図26のファイルシステム構造の情報をUDFフォーマットに従って情報記録媒体上に記録した例を図27～図29に示す。

[0134] 情報記録媒体上の未記録位置管理方法として次のようなものがある。

[0135] • Space Bitmap 方法

Space Bitmap Descriptor 470を用いた、情報記録媒体内記録領域の全論理ブロックに対してビットマップ的に “記録済み” または “未記録” のフラグを立てる。

[0136] • Space Table 方法

Unallocated Space Entry 471の記述方式を用いて Short Allocation Descriptor の列記として未記録の全論理ブロック番号を記録している。

[0137] 本実施形態では説明上、わざと図27～図29に両方を併記しているが、実際には両方が一緒に使われる (情報記録媒体上に記録される) ことはほとんど無く、どちらか一方のみ使われている。

[0138] 図27～図29に記述されている主な Descriptor の内容の概説は以下の通りである。

[0139] • Bagging Extended Area Descriptor 445 ... Volume Recognition Sequenceの開始位置を示す

• Volume Structure Descriptor 446 ... Volumeの内容説明を記述。

[0140] • Boot Descriptor 447 ... ブート時の処理内容を記述。

[0141] • Terminating Extended Area Descriptor 448 ... Volume Recognition Sequenceの終了位置を示す。

[0142] • Partition Descriptor 450 ... パーティション情報 (サイズなど) を示すDVD-RAMでは1Volume当たり1Partitionを原則としている。

[0143] • Logical Volume Descriptor 454 ... 論理ボリュームの内容を記述している。

[0144] • Anchor Volume Descriptor Pointer 458 ... 情報記録媒体記録領域内でのMain Volume Descriptor Sequence 449とMain Volume Descriptor Sequence 467の記録位置を示している。

[0145] • Reserved (all 00h bytes) 459 ~ 465 ... 特定の Descriptor を記録する論理セクタ番号を確保するため、その間に全て “0” を記録した調整領域を持たせている。

[0146] • Reserve Volume Descriptor Sequence 467 ... Main Volume Descriptor Sequence 449に記録された情報のバックアップ領域。

[0147] <<<再生時のファイルデータへのアクセス方法>>>図27～図29に示したファイルシステム情報を用いて例えば File Data H402のデータ内容を再生するための情報記録媒体上のアクセス処理方法について説明する。

下に表示ものが記述されている。

[0121] • Descriptor Tag 413 ... 記述内容の識別子を表し、この場合は “263”

• ICB Tag 414 ... ファイルタイプを示す

ICB Tag 内の File Type=1 は Unallocated Space Entry を意味し、File Type=4 は Directory、File Type=5 は File Data を表している。

[0122] • Allocation Descriptors 列の全長 415 ... 4Bytes で総 Bytes 数を示す。

[0123] <<<File Entry>>>

“<情報記録媒体上のファイルシステム情報記録内容>” で説明した記述文であり、図24に示すように以下に示すものが記述されている。

[0124] • Descriptor Tag 417 ... 記述内容の識別子を表し、この場合は “261”

• ICB Tag 418 ... ファイルタイプを示す → 内容は前述の ICB Tag 414と同じ

• Permissions 419 ... ユーザ別の記録・再生・削除許可情報を示す。主にファイルのセキュリティ確保を目的として使われる。

[0125] • Allocation Descriptors 420 ... 該当ファイルの意味が記録して有る位置をExtent 毎にShort Allocation Descriptor を並べて記述する。

[0126] <<<File Identifier Descriptor>>>

“<情報記録媒体上のファイルシステム情報記録内容>” で説明したようにファイル情報を記述した記述文であり、図25に示すように以下に示すものが記述されている。

[0127] • Descriptor Tag 421 ... 記述内容の識別子を表し、この場合は “257”

• File Characteristics 422 ... ファイルの種類を示し、Parent Directory、Directory、File Data、File Entry、Parent Directory、Directory、File Data、File Entryの削除フラグのどれかを意味する。

[0128] • Information Control Block 423 ... このファイルに対応したF/E位置がLong Allocation Descriptor で記述されている。

[0129] • File Identifier 424 ... ディレクトリ名またはファイル名。

[0130] • Padding 437 ... File Identifier Descriptor 全体の長さを調整するために付加されたダミー領域で、通常は全て “0” が記録されている。

[0131] <<<UDFに使って情報記録媒体上に記録したファイル構造記述例>>>

“<<<UDFの概要>>>” で示した内容について具体的な例を用いて以下に詳細に説明する。

図20 (a) に対して、より一般的なファイルシステム構造例を図26に示す。括弧内は Directory の中身に関する情報または File Data のデータ内容が記録されている情報記録媒体上の論理ブロック番号を示している。

AD(112))を読み取る。

【0155】9) 112番目の論理ブロックにアクセスし、SubDirectory F 430 に関するFile Entry 482を再生し、SubDirectory F 430 の中身に関する情報が記録されている位置 (論理ブロック番号)を読み込む (AD(113))。

【0156】10) 113番目の論理ブロックにアクセスし、SubDirectory F 430 の中身に関する情報を再生し、File Data H 432 に関する File Identifier Descriptor を探す。そしてそこから File Data H 432

に関する File Entry が記録して有る論理ブロック番号 (図27～図29には図示して無いがLAD(114))を読み取る。

【0157】11) 114番目の論理ブロックにアクセスし、File Data H 432 に関するFile Entry 484を再生し、File Data H 432 のデータ内容 489 が記録されている位置を読み取る。

【0158】12) File Data H 432 に関する File Entry 484 内に記述されている論理ブロック番号順に情報記憶媒体から情報を再生して File Data H 432 のデータ内容 489 を読み取る。

【0159】<<<特定のファイルデータ内容変更方法>>>図27～図29に示したファイルシステム情報を用いて例えば File Data H432 のデータ内容を変更する場合のアクセスも含めた処理方法について説明する。

【0160】1) File Data H 432 の変更前後のデータ内容の容量差を求め、その値を2048 Bytes で割り、変更後のデータを記録するのに論理ブロック何個追加使用するかもまたは何個不要になるかを事前に計算しておく。

【0161】2) 情報記録再生装置起動時または情報記憶媒体装着時のブート (Boot) 領域として Volume Recognition Sequence 444 領域内の Boot Descriptor 447 の情報を再生に行く。Boot Descriptor 447 の記述内容に沿ってブート (Boot) 時の処理が始まる。

【0162】3) 特に指定されたブート時の処理が無い場合には、始めに Main Volume Descriptor Sequence 449 領域内の Partition Descriptor 450 を再生し、その中に記述して有る Partition Contents Use 451 の情報を読み取る。この PartitionContents Use 451 (Partition Header Descriptor と呼ぶ) の中に Space Table もしくは Space Bitmap の記録位置が示して有る。

Space Table 位置は Unallocated Space Table 452 の欄に Short Allocation Descriptor の形式で記述されている。(図27～図29の例ではAD(50)) また、Space Bitmap 位置は Unallocated Space Bitmap 453 の欄に Short Allocation Descriptor の形式で記述されている。(図27～図29の例ではAD(0))

4) 3) で読み取った Space Bitmap が記述して有る論理ブロック番号 (0) →アクセスする。Space Bitmap D

escriptor 470 から Space Bitmap 情報を読み取り、未記録の論理ブロックを探し、1) の計算結果分の論理ブロックの使用を登録する (Space Bitmap Descriptor 460 情報の書き換え処理)。

【0163】4') もしくは、3) で読み取った Space Table が記述して有る論理ブロック番号 (50) →アクセスする。Space Table の USE(AD(4), AD(4), ..., AD(4)) 471 から未記録の論理ブロックを探し、1) の計算結果分の論理ブロックの使用を登録する (Space Table 情報の書き換え処理)

実際の処理は "4" か "4'" のうちのどちらから一方の処理を行う。

【0164】5) 次に Main Volume Descriptor Sequence 449 領域内の Logical Volume Descriptor 454 の情報を再生する。

【0165】6) Logical Volume Descriptor 454 の中に Logical Volume Contents Use 455が記述されており、そこに File Set Descriptor 472 が記録して有る位置を示す論理ブロック番号が Long Allocation Descriptor (図21) 形式で記述して有る。(図27～図29の例ではLAD(100)から100番目の論理ブロックに記録して有る。)

7) 100番目の論理ブロック (論理セクタ番号では400番目になる) にアクセスし、File Set Descriptor 472を再生する。その中のRoot Directory 108 473 に Root Directory A 425 に関する File Entry が記録されている場所 (論理ブロック番号) が Long Allocation Descriptor (図21) 形式で記述して有る (図27～図29の例ではLAD(102)から102番目の論理ブロックに記録して有る)。

【0166】8) Root Directory 108 473 の LAD(102)に従い、102番目の論理ブロックにアクセスし、Root Directory A 425 に関するFile Entry 475を再生し、Root Directory A 425 の中味に関する情報が記録されている位置 (論理ブロック番号) を読み込む (AD(103))。

【0167】9) 103番目の論理ブロックにアクセスし、Root Directory A 425 の中味に関する情報を再生する。File Data H 432 は Directory D 428 系列の下に存在するので、Directory D 428 に関する File Identifier Descriptor を探し、Directory Y D 428 に関する File Entry が記録して有る論理ブロック番号 (図27～図29には図示して無いがLAD(110)) を読み取る。

【0168】10) 110番目の論理ブロックにアクセスし、Directory D 428 に関するFile Entry 480を再生し、Directory D 428 の中身に関する情報が記録されている位置 (論理ブロック番号) を読み込む (AD(111))。

【0169】11) 111番目の論理ブロックにアクセス

スし、Directory D 428 の中身に関する情報を再生する。File Data H 432 は SubDirectory F 430 の直接下に存在するので、SubDirectory F 430 に関する File Identifier Descriptor を探し、SubDirectory F 430 に関する File Entry が記録して有る論理ブロック番号 (図27～図29には図示して無いがLAD(112)) を読み取る。

【0170】12) 112番目の論理ブロックにアクセスし、SubDirectory F 430 に関する File Entry 482を再生し、SubDirectory F 430 の中身に関する情報が記録されている位置 (論理ブロック番号) を読み込む (AD(113))。

【0171】13) 113番目の論理ブロックにアクセスし、SubDirectory F 430 の中身に関する情報を再生し、File Data H 432 に関する File Identifier Descriptor を探す。そしてそこから File Data H 432 に関する File Entry が記録して有る論理ブロック番号 (図27～図29には図示して無いがLAD(114))を読み取る。

【0172】14) 114番目の論理ブロックにアクセスし、File Data H 432 に関するFile Entry 484を再生し、File Data H 432 のデータ内容 489 が記録されている位置を読み取る。

【0173】15) 4) か4') で追加登録した論理ブロック番号も加味して変更後の File Data H 432 のデータ内容 489 を記録する。

【0174】<<<特定のファイルデータ/ディレクトリ消去処理方法>>>例として File Data H 432 または SubDirectory F 430 を消去する方法について説明する。

【0175】1) 情報記録再生装置起動時または情報記憶媒体装着時のブート (Boot) 領域として Volume Recognition Sequence 444 領域内の Boot Descriptor 447 の情報を再生に行く。Boot Descriptor 447 の記述内容に沿ってブート (Boot) 時の処理が始まる。

【0176】2) 特に指定されたブート時の処理が無い場合には、始めに Main Volume Descriptor Sequence 449 領域内の Logical Volume Descriptor 454 の情報を再生する。

【0177】3) Logical Volume Descriptor 454 の中に Logical Volume Contents Use 455が記述されており、そこに File Set Descriptor 472 が記録して有る位置を示す論理ブロック番号が Long Allocation Descriptor (図21) 形式で記述して有る。(図27～図29の例ではLAD(100)から100番目の論理ブロックに記録して有る。)

4) 100番目の論理ブロック (論理セクタ番号では400番目になる) にアクセスし、File Set Descriptor 472を再生する。その中のRoot Directory 108 473 に Root Directory A 425 に関する File Entry が記録さ

- れている場所 (論理ブロック番号) が Long Allocation Descriptor (図21) 形式で記述して有る (図27～図29の例ではLAD(102)から102番目の論理ブロックに記録して有る)。
- [0178] 5) Root Directory 108 473 のLAD(102)に従い、102番目の論理ブロックにアクセスし、Root Directory A 425 に関するFile Entry 475 を再生し、Root Directory A 425 の中身に関する情報が記録されている位置 (論理ブロック番号) を読み込む (AD(103))。
- [0179] 6) 103番目の論理ブロックにアクセスし、Root Directory A 425 の中身に関する情報を再生する。File Data H 432 は Directory D 428 系列の下に存在するので、Directory D 428 に関するFile Identifier Descriptor を探し、Directory D 428 に関するFile Entry が記録して有る論理ブロック番号 (図27～図29には図示して無いがLAD(101)) を読み取る。
- [0180] 7) 110番目の論理ブロックにアクセスし、Directory D 428 に関するFile Entry 480 を再生し、Directory D 428 の中身に関する情報が記録されている位置 (論理ブロック番号) を読み込む (AD(111))。
- [0181] 8) 111番目の論理ブロックにアクセスし、Directory D 428 の中身に関する情報を再生する。File Data H 432 は SubDirectory F 430 の直下に存在するので、SubDirectory F 430 に関するFile Identifier Descriptor を探す。
- [0182] SubDirectory F 430 を消去する場合には、SubDirectory F 430 に関するFile Identifier Descriptor 内のFile Characteristics 422 (図25) に“ファイル削除フラグ”を立てる。SubDirectory F 430 に関するFile Entry が記録して有る論理ブロック番号 (図27～図29には図示して無いがLAD(112)) を読み取る。
- [0183] 9) 112番目の論理ブロックにアクセスし、SubDirectory F 430 に関するFile Entry 482 を再生し、SubDirectory F 430 の中身に関する情報が記録されている位置 (論理ブロック番号) を読み込む (AD(113))。
- [0184] 10) 113番目の論理ブロックにアクセスし、SubDirectory F 430 の中身に関する情報を再生し、File Data H 432 に関するFile Identifier Descriptor を探す。
- [0185] File Data H 432 を消去場合には、File Data H 432 に関するFile Identifier Descriptor 内のFile Characteristics 422 (図25) に“ファイル削除フラグ”を立てる。さらにそこからFile Data H 432 に関するFile Entry が記録して有る論理ブロック番号 (図27～図29には図示して無いがLAD
- (114)) を読み取る。
- [0186] 11) 114番目の論理ブロックにアクセスし、File Data H 432 に関するFile Entry 484 を再生し、File Data H 432 のデータ内容 489 が記録されている位置を読み取る。
- [0187] File Data H 432 を消去する場合には、以下の方法でFile Data H 432 のデータ内容 489 が記録されていた論理ブロックを解放する (その論理ブロックを未記録状態に登録する)。
- [0188] 12) 次に Main Volume Descriptor Sequence 449 領域内の Partition Descriptor 450 を再生し、その中に記述して有る Partition Contents Use 451 の情報を読み取る。この Partition Contents Use 451 (Partition Header Descriptor と呼ぶ) の中にSpace Table もしくは Space Bitmap の記録位置が示して有る。
- [0189] Space Table 位置は Unallocated Space Identifier 452 の欄に Short Allocation Descriptor の形式で記述されている。 (図27～図29の例ではAD(50)) また、Space Bitmap 位置は Unallocated Space Identifier 453 の欄に Short Allocation Descriptor の形式で記述されている。 (図27～図29の例ではAD(0))
- [0190] 13) 12) で読み取った Space Bitmap が記述して有る論理ブロック番号 (0) へアクセスし、11) の結果得られた“解放する論理ブロック番号”を SpaceBitmap Descriptor 470 に書き換える。
- [0190] 13') もしくは、12) で読み取った Space Table が記述して有る論理ブロック番号 (50) へアクセスし、11) の結果得られた“解放する論理ブロック番号”を Space Table に書き換える。
- [0191] 実際の処理は“13)”か“13')”のうちのどちらか一方の処理を行う。
- [0192] File Data H 432 を消去する場合には、12) 10) ～11) と同じ手順を繰り返して File Data Identifier 433 のデータ内容 490 が記録されている位置を読み取る。
- [0193] 13) 次に Main Volume Descriptor Sequence 449 領域内の Partition Descriptor 450 を再生し、その中に記述して有る Partition Contents Use 451 の情報を読み取る。この Partition Contents Use 451 (Partition Header Descriptor と呼ぶ) の中にSpace Table もしくは Space Bitmap の記録位置が示して有る。
- [0194] Space Table 位置は Unallocated Space Identifier 452 の欄に Short Allocation Descriptor の形式で記述されている。 (図27～図29の例ではAD(50)) また、Space Bitmap 位置は Unallocated Space Identifier 453 の欄に Short Allocation Descriptor の形式で記述されている。 (図27～図29の例ではAD

- (0))
- 14) 13) で読み取った Space Bitmap が記述して有る論理ブロック番号 (0) へアクセスし、11) と12) の結果得られた“解放する論理ブロック番号”を Space Bitmap Descriptor 470 に書き換える。
- [0195] 14') もしくは、13) で読み取った Space Table が記述して有る論理ブロック番号 (50) へアクセスし、11) と12) の結果得られた“解放する論理ブロック番号”を Space Table に書き換える。
- [0196] 実際の処理は“14)”か“14')”のうちのどちらか一方の処理を行う。
- [0197] <<<ファイルデータ/ディレクトリの追加処理>>>例として SubDirectory F 430 の下に新たにファイルデータもしくはディレクトリを追加する時のアクセス・追加処理方法について説明する。
- [0198] 1) ファイルデータを追加する場合には追加するファイルデータ内容の容量を調べ、その値を2048 Bytes で割り、ファイルデータを追加するために必要な論理ブロック数を計算しておく。
- [0199] 2) 情報記録再生装置起動時または情報記録媒体装着時のブート (Boot) 領域として Volume Recognition Sequence 444 領域内の Boot Descriptor 447 の情報を再生し、Boot Descriptor 447 の記述内容に沿ってブート (Boot) 時の処理が始まる。
- [0200] 3) 特に指定されたブート時の処理が無い場合には、始めに Main Volume Descriptor Sequence 449 領域内の Partition Descriptor 450 を再生し、その中に記述して有る Partition Contents Use 451 (Partition Header Descriptor と呼ぶ) の中にSpace Table もしくは Space Bitmap の記録位置が示して有る。
- [0201] Space Table 位置は Unallocated Space Identifier 452 の欄に Short Allocation Descriptor の形式で記述されている。 (図27～図29の例ではAD(50)) また、Space Bitmap 位置は Unallocated Space Identifier 453 の欄に Short Allocation Descriptor の形式で記述されている。 (図27～図29の例ではAD(0))
- 4) 3) で読み取った Space Bitmap が記述して有る論理ブロック番号 (0) へアクセスする。Space Bitmap Descriptor 470 から Space Bitmap 情報を読み取り、未記録の論理ブロックを探し、1) の計算結果分の論理ブロックの使用を登録する (Space Bitmap Descriptor 470 情報の書き換え処理)。
- [0202] 4') もしくは、3) で読み取った Space Table が記述して有る論理ブロック番号 (50) へアクセスする。Space Table の USE(40(*), AD(*), ..., AD(*)) 471 から未記録の論理ブロックを探し、1) の計算結果分の論理ブロックの使用を登録する (Space Table 情報の書き換え処理) 実際の処理は“4)”か“4')”
- 50 情報が記録されている位置 (論理ブロック番号) を読み
- のうちのどちらか一方の処理を行う。
- [0203] 5) 次に Main Volume Descriptor Sequence 449 領域内の Logical Volume Descriptor 454 の情報を再生する。
- [0204] 6) Logical Volume Descriptor 454 の中に Logical Volume Contents Use 455 が記述されており、そこに File Set Descriptor 472 が記録して有る位置を示す論理ブロック番号が Long Allocation Descriptor (図21) 形式で記述して有る。 (図27～図29の例ではLAD(100)から100番目の論理ブロックに記録して有る)。
- 7) 100番目の論理ブロック (論理セクタ番号では400番目になる) にアクセスし、File Set Descriptor 472 を再生する。その中のRoot Directory 108 473 に Root Directory A 425 に関するFile Entry が記録されている場所 (論理ブロック番号) が Long Allocation Descriptor (図21) 形式で記述して有る (図27～図29の例ではLAD(102)から102番目の論理ブロックに記録して有る)。
- [0205] 8) Root Directory 108 473 のLAD(102)に従い、102番目の論理ブロックにアクセスし、Root Directory A 425 に関するFile Entry 475 を再生し、Root Directory A 425 の中身に関する情報が記録されている位置 (論理ブロック番号) を読み込む (AD(103))。
- [0206] 9) 103番目の論理ブロックにアクセスし、Root Directory A 425 の中身に関する情報を再生する。
- [0207] Directory D 428 に関するFile Identifier Descriptor を探し、Directory D 428 に関するFile Entry が記録して有る論理ブロック番号 (図27～図29には図示して無いがLAD(110)) を読み取る。
- [0208] 10) 110番目の論理ブロックにアクセスし、Directory D 428 に関するFile Entry 480 を再生し、Directory D 428 の中身に関する情報が記録されている位置 (論理ブロック番号) を読み込む (AD(111))。
- [0209] 11) 111番目の論理ブロックにアクセスし、Directory D 428 の中身に関する情報を再生する。
- [0210] SubDirectory F 430 に関するFile Identifier Descriptor を探し、SubDirectory F 430 に関するFile Entry が記録して有る論理ブロック番号 (図27～図29には図示して無いがLAD(112)) を読み取る。
- [0211] 12) 112番目の論理ブロックにアクセスし、SubDirectory F 430 に関するFile Entry 482 を再生し、SubDirectory F 430 の中身に関する情報が記録されている位置 (論理ブロック番号) を読み込む (AD(113))。

込む (AD(113))。

【0212】13) 113番目の論理ブロックにアクセスし、Subdirectory F 430 の中身に関する情報内に新たに追加するファイルデータもしくはディレクトリ File Identifier Descriptor を登録する。

【0213】14) または14') で登録された論理ブロック番号位置にアクセスし、新たに追加するファイルデータもしくはディレクトリに関する File Entry を記録する。

【0214】15) 14) の File Entry 内の Short Allocation Descriptor に示した論理ブロック番号位置にアクセスし、追加するディレクトリに関する Parent Directory の File Identifier Descriptor もしくは追加するファイルデータのデータ内容を記録する。

【0215】続いて、映像情報の記録時の連続性及びCDAについて補足説明する。これらに関しては、特開2000-112673に記載されている。

【0216】映像情報は従来のコンピュータ情報となり、記録時の連続性の保証が必須条件となる。以下にこの記録時の連続性を阻害する理由の説明と、記録時の連続性を保証する方法について説明する。

【0217】外部から送られてくる映像情報はバッファメモリ (半導体メモリ) に一時保管される。粗アクセスと密アクセス動作により光学ヘッドが情報記憶媒体上の記録位置へ到達すると、上記バッファメモリに一時保管された映像情報が光学ヘッドの読出しで情報記憶媒体上に記録される。バッファメモリから光学ヘッドへ送られる映像情報の転送レートをここでは物理転送レート (PTR: Physical Transmission Rate) と定義する。外部からバッファメモリへ転送される映像情報の転送レートの平均値をシステム転送レート (STR: System Transmission Rate) とここで定義する。一般には物理転送レートPTRとシステム転送レートSTRとは異なる値になっている。

【0218】情報記憶媒体上の異なる場所順に順次映像情報を記録するには光学ヘッドの集光スポット位置を移動させるアクセス操作が必要となる。大きな移動に対しては光学ヘッド全体を動かす粗アクセスを行い、微少距離の移動にはレーザ集光用の対物レンズのみを動かす密アクセスを行う。

【0219】続いて、外部から転送されてくる映像情報に対して光学ヘッドのアクセス制御を行ないながら情報記憶媒体上の所定位置に順次映像情報を記録する場合のバッファメモリ内に一時的に保存される映像情報量の時間的推移を示す。一般にシステム転送レートSTRより物理転送レートPTRの方が速いので映像情報記録時間の期間ではバッファメモリ内に一時的に保存される映像情報量は減少し続ける。バッファメモリ内に一時保管される映像情報量が“0”になる。その時には連続的に転送されてくる映像情報はバッファメモリ内に一時

間をJATaとし、各アクセスの映像情報記録時間をDWTi (Data Write Time)、n回アクセス後の平均回転待ち時間をMWTaとする。

【0220】n回アクセスした場合の全アクセス期間で外部からバッファメモリへ転送される映像情報データ量は、

$$STR \times (\Sigma (SATi + JATi) + MWTi) \times n \times (SATa + JATa + MWTa) \quad (1)$$

となる。

【0221】また、n回アクセスして映像情報記録時に必要となる、

$$(PTR - STR) \times \Sigma DWTi \times n \cdot DWTa \quad (2)$$

となる。

★映像情報記録時の連続性が確保される。

【0222】式(1)と式(2)の間で、下記式

$$(PTR - STR) \times n \cdot DWTa \geq STR \times n \times (SATa + JATa + MWTa) \quad (3)$$

の関係にあるときに、外部システム側から見た映像情報の転送レートをTaとすると、

$$Ta = SATa + JATa + MWTa \quad (4)$$

ここで1回のアクセスに必要な平均時間をTaとすると、

$$(PTR - STR) \times DWTa \geq STR \times Ta \quad (5)$$

と変形される。

◆す。一回のアクセス後に情報記憶媒体上に連続記録するデータ領域が、上記したCDAである。(5)式から、

$$DWTa \geq STR \times Ta / (PTR - STR) \quad (6)$$

と変形できる。

【0231】CDAサイズを示すCDAは、

$$CDA \geq STR \times PTR \times Ta / (PTR - STR) \quad (8)$$

となる。式(8)から連続記録を可能にするためのCDAサイズの下限値を規定できる。

【0232】粗アクセス、密アクセスに必要な時間は情報記録再生装置の性能により大きく異なる。今仮に、SATA 200 ms (9)

を仮定する。前述のように例えばMWTa 18 ms、JATa 5 ms を計算に使う。

【0233】2.6GB DVD-RAM では、

$$PTR = 11.09 Mbps \quad (10)$$

である。MPEG2の平均転送レートが、

$$STR = 4 Mbps \quad (11)$$

の場合には上記の数値を式(8)に代入すると、

$$CDA \geq 1.4 Mbits \quad (12)$$

を得る。また別の見解もありとして、

$$SATa + JATa + MWTa = 1.5 秒 \quad (13)$$

とした場合には式(8)から、

$$CDA \geq 9.4 Mbits \quad (14)$$

となる。また録画DVDの規格上では、MPEG2の最大転送レートとして、

$$STR = 8 Mbps \quad (15)$$

以下になるように規定しているので、式(15)の値を式(8)に代入すると、

$$CDA \geq 4.3.2 Mbits = 5.4 Mbytes \quad (16)$$

【図2】図1に示す共存方法において、UDF論理アドレス

＊e Motor Wait Time)とし、n回アクセス後の平均回転待ち時間をMWTaとする。

【0226】n回アクセスした場合の全アクセス期間で外部からバッファメモリへ転送される映像情報データ量は、

$$STR \times (\Sigma (SATi + JATi) + MWTi) \times n \times (SATa + JATa + MWTa) \quad (1)$$

となる。

【0227】また、n回アクセスして映像情報記録時に必要となる、

$$(PTR - STR) \times \Sigma DWTi \times n \cdot DWTa \quad (2)$$

となる。

★映像情報記録時の連続性が確保される。

【0229】

$$(PTR - STR) \times DWTa \geq STR \times n \times (SATa + JATa + MWTa) \quad (3)$$

ここで1回のアクセスに必要な平均時間をTaとすると、

$$Ta = SATa + JATa + MWTa \quad (4)$$

と変形される。

◆す。一回のアクセス後に情報記憶媒体上に連続記録するデータ領域が、上記したCDAである。(5)式から、

$$DWTa \geq STR \times Ta / (PTR - STR) \quad (6)$$

と変形できる。

【0231】CDAサイズを示すCDAは、

$$CDA \geq STR \times PTR \times Ta / (PTR - STR) \quad (8)$$

となる。式(8)から連続記録を可能にするためのCDAサイズの下限値を規定できる。

【0232】粗アクセス、密アクセスに必要な時間は情報記録再生装置の性能により大きく異なる。今仮に、SATA 200 ms (9)

を仮定する。前述のように例えばMWTa 18 ms、JATa 5 ms を計算に使う。

【0233】2.6GB DVD-RAM では、

$$PTR = 11.09 Mbps \quad (10)$$

である。MPEG2の平均転送レートが、

$$STR = 4 Mbps \quad (11)$$

の場合には上記の数値を式(8)に代入すると、

$$CDA \geq 1.4 Mbits \quad (12)$$

を得る。また別の見解もありとして、

$$SATa + JATa + MWTa = 1.5 秒 \quad (13)$$

とした場合には式(8)から、

$$CDA \geq 9.4 Mbits \quad (14)$$

となる。また録画DVDの規格上では、MPEG2の最大転送レートとして、

$$STR = 8 Mbps \quad (15)$$

以下になるように規定しているので、式(15)の値を式(8)に代入すると、

$$CDA \geq 4.3.2 Mbits = 5.4 Mbytes \quad (16)$$

【図2】図1に示す共存方法において、UDF論理アドレス

＊e Motor Wait Time)とし、n回アクセス後の平均回転待ち時間をMWTaとする。

【0226】n回アクセスした場合の全アクセス期間で外部からバッファメモリへ転送される映像情報データ量は、

$$STR \times (\Sigma (SATi + JATi) + MWTi) \times n \times (SATa + JATa + MWTa) \quad (1)$$

となる。

【0227】また、n回アクセスして映像情報記録時に必要となる、

$$(PTR - STR) \times \Sigma DWTi \times n \cdot DWTa \quad (2)$$

となる。

★映像情報記録時の連続性が確保される。

【0229】

$$(PTR - STR) \times DWTa \geq STR \times n \times (SATa + JATa + MWTa) \quad (3)$$

ここで1回のアクセスに必要な平均時間をTaとすると、

$$Ta = SATa + JATa + MWTa \quad (4)$$

と変形される。

◆す。一回のアクセス後に情報記憶媒体上に連続記録するデータ領域が、上記したCDAである。(5)式から、

$$DWTa \geq STR \times Ta / (PTR - STR) \quad (6)$$

と変形できる。

【0231】CDAサイズを示すCDAは、

$$CDA \geq STR \times PTR \times Ta / (PTR - STR) \quad (8)$$

となる。式(8)から連続記録を可能にするためのCDAサイズの下限値を規定できる。

【0232】粗アクセス、密アクセスに必要な時間は情報記録再生装置の性能により大きく異なる。今仮に、SATA 200 ms (9)

レス空間上で各種ファイルの配置設定方法を比較した図である。

【図 3】図 1 に示す共有方法において、UDF 論理アドレス空間上での DVD オブジェクトファイルの配置設定方法の一例を示す図である。

【図 4】図 1 に示す共有方法において、UDF 論理アドレス空間上での CE 領域に格納したファイルシステム空間を示すファイルの配置設定方法の一例を示す図である。

【図 5】図 1 に示す共有方法における各種ファイルと疑似ファイルの説明するための図である。

【図 6】CE_FILE_AREA、CEFF のファイルサイズの拡大(縮小)を説明するための図である。

【図 7】UDF による従来のボリューム & ファイルストラクチャの読み出し手順を示すフローチャートである。

【図 8】図 1 に示す共有方法を採用したときのボリューム & ファイルストラクチャの読み出し手順を示すフローチャートである。

【図 9】CE_FILE_AREA、CEFF の割り当てと、WRITE/READ 処理方法を説明するための図である。

【図 10】Trigger ファイルのデータ構造の一例を示す図である。

【図 11】Trigger ファイルにより CE ファイル位置を識別する処理を示すフローチャートである。

【図 12】Trigger ファイルの先頭セクタを読み出せない場合の処理を示すフローチャートである。

【図 13】Trigger ファイルを活用し CE ファイルを追加又は削除する処理を示すフローチャートである。

【図 14】図 1 に示す共有方法を実現するためのシステムの概略構成を示す図である。

【図 15】図 1 に示す共有方法とは異なる共有方法を示す図である。

【図 16】図 15 に示す共有方法における CE 領域に格納したファイルシステム空間上での UDF 論理アドレス空間を示すファイル配置を示す図である。

【図 17】Trigger ファイルで管理される CE_FILE_AREA、CEFF の記録処理を示すフローチャートである。

【図 18】Trigger ファイルで管理される CE_FILE_AREA、CEFF の追加処理を示すフローチャートである。

ャートである。
【図 19】Trigger ファイルで管理される CE_FILE_AREA、CEFF のサイズ縮小処理を示すフローチャートである。

【図 20】階層化されたファイルシステム構造と情報記憶媒体上へ記録された情報内容との間の基本的な関係の概念を示した概念説明図である。

【図 21】Long Allocation Descriptor (Extent) の位置を示す大きいサイズの記述文) の記述内容説明図である。

【図 22】Short Allocation Descriptor (Extent) の位置を示す小さいサイズの記述文) の記述内容説明図である。

【図 23】Unallocated Space Entry (未記録な Extent) の情報記憶媒体上の位置に関する特設登録用の記述文) の記述内容説明図である。

【図 24】File Entry (File の属性と File の記録位置の情報登録に関する記述文) の記述内容を一部抜粋した内容説明図である。

【図 25】File Identifier Descriptor (File の名前と対応した FFE の記録位置に関する記述文) の記述内容を一部抜粋した内容説明図である。

【図 26】ファイルシステム構造の一例を示す図である。

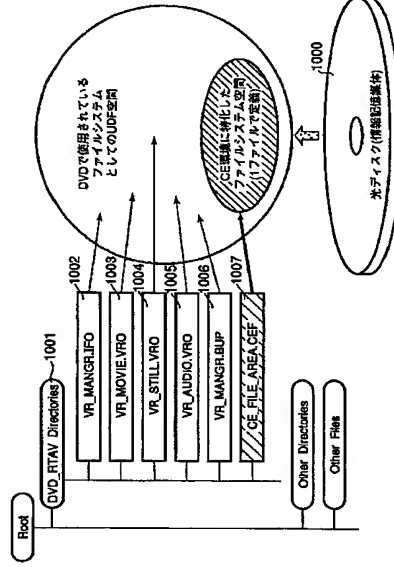
【図 27】UDF に従って情報記憶媒体上にファイルシステムを記録した例を示す図の一部(その 1)である。

【図 28】UDF に従って情報記憶媒体上にファイルシステムを記録した例を示す図の一部(その 2)である。

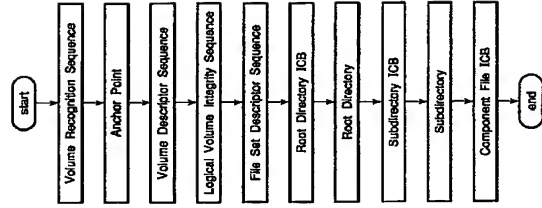
【図 29】UDF に従って情報記憶媒体上にファイルシステムを記録した例を示す図の一部(その 3)である。

【符号の説明】
1001...DVD_RTAV Directories
1002...VR_MANGR. IFO
1003...VR_MOVIE. VRO
1004...VR_STILL. VRO
1005...VR_AUDIO. VRO
1006...VR_MANGR. BUP
1007...CE_FILE_AREA. CEF
2001...TOC_FILE. IFO
2002...AV_MANG. MNG
2004...AV_FILE_01.MPG
2005...AV_FILE_02.MPG
2006...UDF_FILE_AREA. UDF

【図 1】



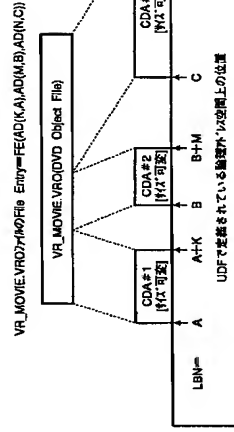
【図 7】



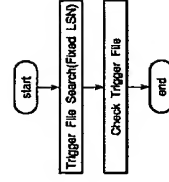
【図 2】

比較項目	PO File 管理File	DVD Object File	CE領域に格納したファイルが占める空間
CDA#1 (記録時の書き出し単位)	CDA設定無し 2KB以上の可変	2KB以上の可変	4KB固定
CDA境界位置の保持	CDA設定無し	境界位置を保持しない	常にCDA境界位置を保持する
部分消去時の復旧性	2KB単位	2KB単位	4KB固定 (CDA単位の解除)
CDAとExtentの関係	CDA設定無し	1個以上のExtentで1個のCDAを構成	nCDA=1-Extent(n=1)を保持
図1での該当File	VR_MANGR. IFO VR_MANGR. BUP	VR_MOVIE. VRO VR_STILL. VRO VR_AUDIO. VRO	CE_FILE_AREA. CEF

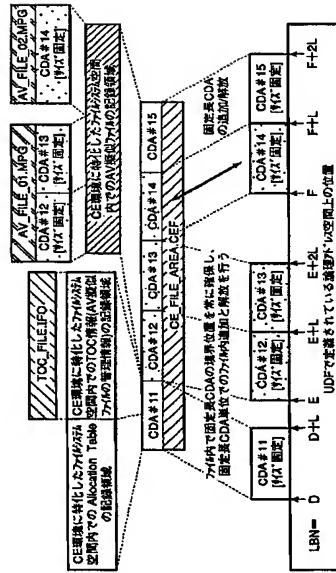
【図 3】



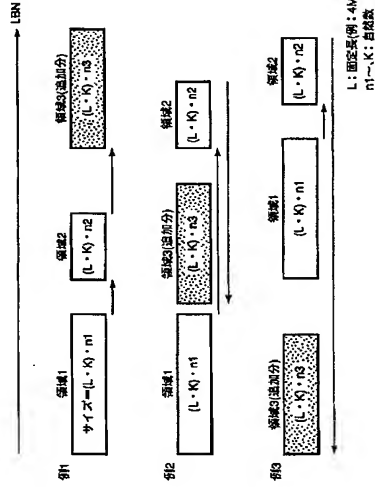
【図 8】



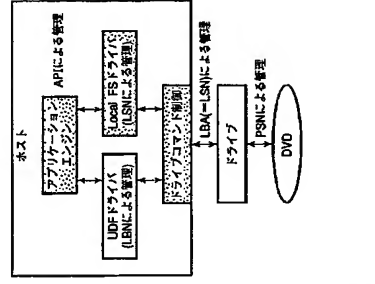
【図4】



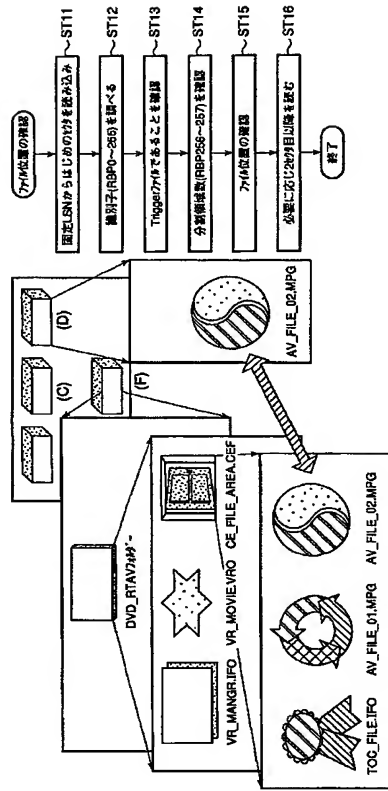
【図6】



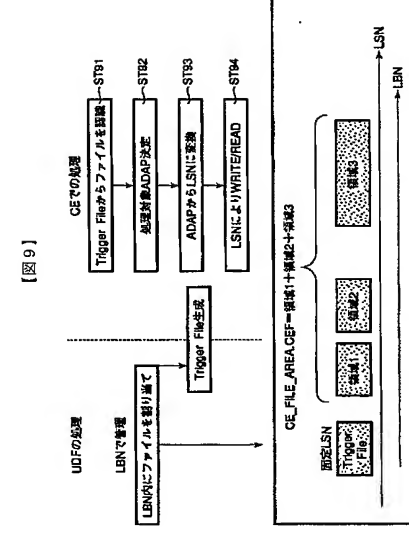
【図14】



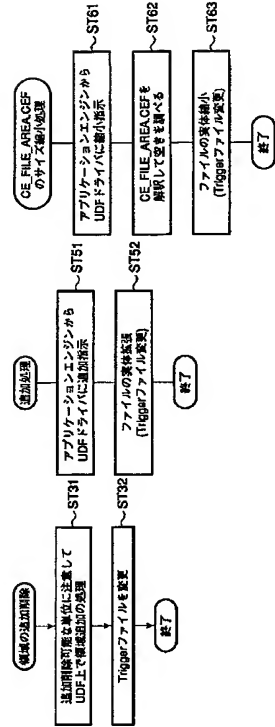
【図11】



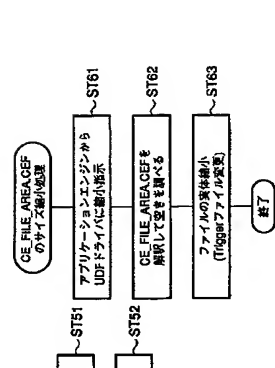
【図9】



【図13】



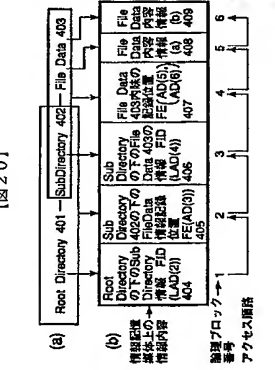
【図18】



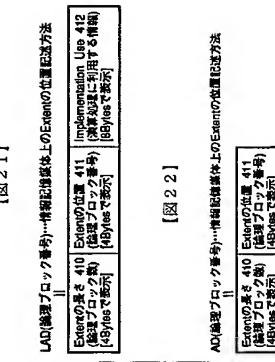
【図19】



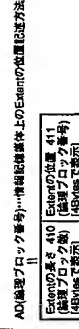
【図20】



【図21】



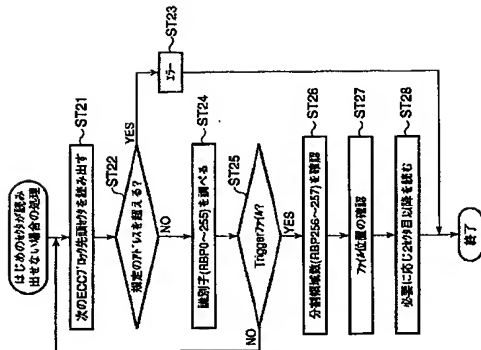
【図22】



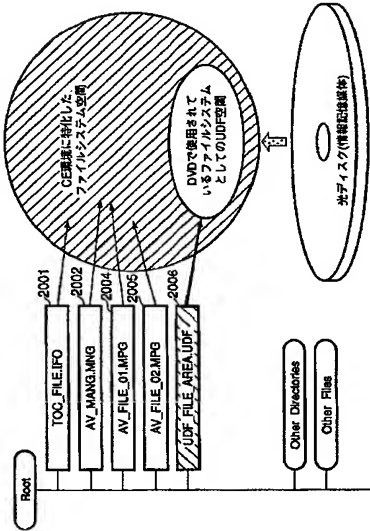
【图10】

RBP	Length	Contents
0 ~ 255	256	Triggerファイル識別子
256 to 257	2	予約領域
258 to 259	2	固定式GDA(車載セクタ数)
260 to 261	2	標準式GDA(除故障発生直前直後GDA単位0x0000) 例: GDA値を16進でとるなら, 0x010h
262 to 265	4	1番目の領域識別LSN
266 to 269	4	1番目の領域識別(セクタ数)
270 to 273	4	2番目の領域識別LSN
274 to 277	4	2番目の領域識別(セクタ数)
...
N to N+3	4	n番目の領域識別LSN
N+4 to N+7	4	n番目の領域識別(セクタ数)
N+8 to LN	Reserved	LN間→ECCプロシユアの最終的RBP
LN+1 to	上記RBP→LNの内容を複数回繰り返す 順次に扱う(する)

【例 12】



【图15】



【23】

USE(AD(*),AD(*)...AD(*))

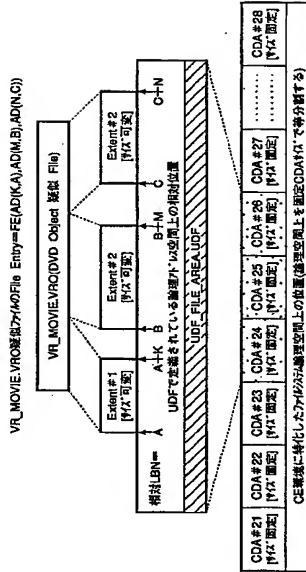
Descriptor	ICB Info	Allocation Descriptors	Allocation Descriptors
ファイル名 (Type=283)	ファイルの タイプを示す (Type=1)	列の全長 (Bytes長)	表Eidantの情報記述媒体上 位置
記述内容の 識別子 413	414	415	(情報記述媒体上の処理) ブロック番号
1[68]bytes	[20B]bytes		を並べて列記号 (AD*)AD*(*)...AD*(*) 416

【图24】

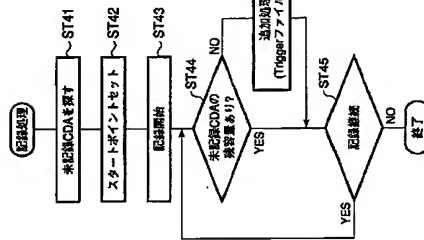
$$FE(AD(*), AD(*) \dots AD(*))$$

Decipher	ICB Tag	Permissions	File/Descriptors
Tag(=281)	ファイル名	ユーザの属性	ファイルの属性と権限
Tag(=282)	ファイルの所有者	記録・属性、	位置
Tag(=417)	属性(=45)	許可情報 418	(情報記述属性上の整理)
168bytes	418	[32B/byte]	(ブロッコック)を を並べて列記 (AD)*AD(*)...AD(*)
	[20B/byte]		

【图16】



【图17】



【25】

FID/LAD(論理ブロック番号))

Descriptor Tag=257 記述用子 421 識別子 421 (16Bytes)	File Characteristics ファイルの識別 を示す 422 (1Byte)	Information Control Block 対応したFEの 記録位置 423 (LAD*)	File Identifier ディレクトリ 名からファイル データ名 424	Padding ダメージ 領域 (000h) 437
---	--	---	--	--

【✕27】

LEN	LEN	Structure 441	Descriptors 442	Contents 443
0-15			Reserved 459 (all 00h bytes)	
16		Volume Partition Sequence 444	Beginning Ext. Area Descr. 445 Volume Structure Descrp. 446 Boot Descriptor 447 Terminating Ext. Area D. 448	VRS開始位置 DISC内容説明 Boot開始位置 VRS終了位置
17			Reserved 460 (all 00h bytes)	
18			保留	
19				
~31				
32~				
34		Main Volume Descriptor Sequence 449	Partition Descriptor 450 Partition Contents Use 451 Unallocated Space Descriptor 452 ADIO 453 Unallocated Space Bitmap 453 ADIO	Space Table の開始位置 Space Bitmap の開始位置
35			Logical Volume Descriptor 454 Logical Volume Cont. Use 455 LUN(103)	File Set Descriptorの 開始位置
~47			保留	
~63			保留	
~255			Reserved 461 (all 00h bytes)	
256		First Anchor Point 456	Anchor Volume Descriptor Pointer 456	
~271			Reserved 462 (all 00h bytes)	

【图26】

